UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P O Box 1450
Alexandria, Virginia 22313-1450
www uspto gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/529,638	11/04/2005	Josef Dietl	09282.0058-00	1384

60668 7590 02/16/2011
SAP / FINNEGAN, HENDERSON LLP
901 NEW YORK AVENUE, NW
WASHINGTON, DC 20001-4413

EXAMINER	
KRETZMER, ERIKA A	

ART UNIT	PAPER NUMBER
2192	

MAIL DATE	DELIVERY MODE
02/16/2011	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	**Application No.** 10/529,638	**Applicant(s)** DIETL ET AL.	
	Examiner Erika Kretzmer	**Art Unit** 2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on _25 May 2010_.

2a)□ This action is **FINAL**. 2b)☒ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) _1-8 and 10-21_ is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) _1-8 and 10-21_ is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on _5/25/2010_ is/are: a)☒ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date _____

5) □ Notice of Informal Patent Application

6) □ Other: _____.

DETAILED ACTION

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114.

2. The following is a non-final action in response to applicant's amendment and response 5/25/2010, responding to the 2/25/2010 office action provided in rejection of claims 1-8 and 10-21. Claims 1, 3, 5, 12, 14, and 16 have been amended. Claims 1-8 and 10-21 are pending and are addressed in this office action. New grounds of rejection are presented in view of the newly presented limitation(s).

Response to Arguments

3. Applicant's arguments filed5/25/2010 in particular pages 12-17, have been fully considered.

4. With respect to the **objection to the drawings**, the amendments to the drawings overcome these objections and they are withdrawn.

5. With respect to the obviousness-type **double patenting rejection**, a terminal disclaimer was filed on 5/25/2010. The rejection is withdrawn.

6. With respect to the previous **rejection of the claims under 35 USC 112, second paragraph**, the amendments to the claims overcome this rejection and it is withdrawn. The amendments to the claims give rise to new issues under 35 USC 112, and new rejections are presented herein.

7. With respect to the **rejection of claims under 35 USC 103(a),** the amendments to the independent claims (1 and 16) overcome the rejections presented in the previous Office Action. New rejections are presented herein. Applicant's arguments have been considered but are moot in view of the new ground(s) of rejection.

Response to Amendment

8. Applicant's amendment to the claims filed 5/25/2010 present substantial new limitations, which raises the issue of new matter under 25 USC 112, first paragraph. Examiner is not readily able to find support for each of the new limitations in the specification as amended or as originally filed. Applicant is requested to point out where in the specification each new limitation finds support, in order to avoid the issue of new matter under 25 USC 112, first paragraph.

Specification

9. The amendment to the specification filed 5/25/2010 has been entered.

10. The amendment to the specification filed 11/25/2009 has been entered because it finds support at least in the original claims. Examiner interprets further references to an "actual" implementation in the specification to be equivalent to the definition module and its class (such as "the actual implementation of the provider class" (page 2 lines 15-16), "the implementation" (page 2 line 30) and "implementation class description" (page 3 lines 25-26)). This interpretation is consistent with the amendments to the specification as filed.

11. The original specification filed 3/31/2005 was published as pre-grant publication US 2006/0248538 A1. The original specification was amended by Applicant at the time of filing on 3/31/2005, and further amended on 1/25/2009.

12. The specification is accepted as amended.

Drawings

13. Original drawing 1 was received on March 31, 2005. Amended drawing 1 and new drawing 2 were received on 5/25/2010. The drawings are accepted.

Claim Rejections - 35 USC § 112

14. The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

15. Claims 1-8 and 10-21 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claims contain subject matter which was not described in the original specification, filed 3/31/2005, in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

16. Claim 1 has been amended to recite "determining whether the interface includes an additional method that is called during runtime execution of the computer program" (lines 18-19). The Examiner is unable to find support for this limitation in the portion of the specification cited by Applicant (page 2, line 17 through page 3 lines 4, and page 4 lines 22-28, as noted in Applicant's remarks filed 5/25/2010, page 12). There is no description of an interface including an additional method that is called during runtime execution in the disclosure as filed, nor does the specification reasonably convey or suggest that the interface includes such a method. Instead, the closest description of this limitation in the specification is on page 2, lines 28-31: "if an interface promises the availability of a certain method but the method is not present in the implementation, a customer of that method would fail at latest at run time." Under the principles of compact prosecution, examiner treats this limitation as pertaining to the interface includes a method that is not present in the implementation (specification, page 2 lines 28-31).

17. Claim 12 recites a similar limitation and is rejected for the same reasons. Claims 2-9, 10-11, and 13-21 are rejected because they depend from rejected claims.

18. Claim 3 has been amended to recite "the class having the object is converted into a new class" (lines 2-3). Applicant notes that support for the new amendments may be found in page 2, line 17 through page 3 lines 4 and page 4 lines 22-28 of the original specification (Applicant's remarks filed 5/25/2010, page 12). However, there is no description of a "new class" the disclosure as filed, nor does the specification reasonably convey that the interface should include such conversion of one class into another new class. Under the principles of compact prosecution, examiner treats this limitation as identical to claim 3 as originally filed. The Examiner understands the original claim 3 to be described in the specification at page 4 lines 4-10, particularly: "the generation of executable programs in script languages, such as for example JavaScript is done by first generating an intermediate code, using a language that supports classes and interfaces."

19. Claim 14 recites a similar limitation and is rejected for the same reasons. Claims 4, 5, 15, and 16 are rejected because they depend from rejected claims.

20. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

21. As noted in a prior Office Action, in view of the amended specification paragraphs received 11/25/2009, the Examiner interprets further references to an "actual" implementation in the specification, such as "the actual implementation of the provider class" (page 2 lines 15-16), "the implementation" (page 2 line 30) and "implementation class description" (page 3 lines 25-26) to be equivalent to the definition instructions and its class. This interpretation is consistent with the amendments to the specification as filed.

22. Claims 1-8 and 10-21 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

23. Claim 1 includes "a class having an object with a runtime function" (lines 5-6) and "a provider class" (lines 7-8). Claim 1 further recites "the class" in line 16 and in line 17. The limitation "the class" is indefinite because it is unclear which "class" in the computer program is intended to be further described by the limitations of lines 16 and 17. Examiner is able to find support for "determining whether the object is relying on a feature of the provider class that is not promised by the provider class" (specification, page 2 lines 11-14). Examiner is unable to find support for the alternate interpretation in the specification. Under the principles of compact prosecution, examiner treats the limitation as being equivalent to the supported limitation.

24. Claim 12 recites a similar limitation and is rejected for the same reasons. Claims 2-9, 10-11, and 13-21 are rejected because they depend from rejected claims.

Claim Rejections - 35 USC § 103

25. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

26. Claims 1-8 and 10-21 are rejected under 35 U.S.C. 103(a) as being unpatentable over "Java and XML Data Binding" (McLaughlin, 2002) in view of "The Java Language Specification, Second Edition" (Gosling et al., 2000) and in view of Lucas et al. (US 6,754,884 B1).

27. Newly added amendments to the claims are underlined to aid the reader.

Claim 1

McLaughlin teaches *a computer-implemented method for validating computer code* (see at least section 3.1.4, Figure 3-1 validating Binding schema against DTD). McLaughlin further teaches *the method being executed by a computer* (see at least page 13, sidebar 1, "at runtime").

McLaughlin further teaches *providing a computer program by defining at least one set of definition instructions* ("binding schema") and *at least one set of implementation instructions* ("constraint model") (see at least section 3.1). McLaughlin further teaches *the set of definition instructions* ("binding schema") *includes a class having an object with a runtime function* (see at least section 3.3, particularly: "Once you've got your constraints ... you're ready to create a binding schema for your classes. This will instruct the class generation tool to generate classes, to use a specific Java package, to use collections, and a variety of other options.", example 3-4 "Modified binding schema for movies database", particularly, "type='class'" and "The result of this addition is apparent in the Movies class, which has multiple Movie subobjects"). McLaughlin further teaches *the set of implementation instructions* ("constraint model") *includes an interface* ("element") *having a method* ("attribute", "ATTLIST") (see at least section 3.2, "Creating the constraints", particularly "a set of constraints ready to generate classes from" and example 3-1 "ELEMENT movies" and "ATTLIST movies version"). McLaughlin further teaches <u>*the class having the object is associated with a provider class*</u> (see at least section 3.3, particularly "The result of this addition is apparent in the Movies class, which has multiple Movie subobjects"). The object ("Movie subobjects") is relying on features of a provider class (Movie class).

McLaughlin further teaches *validating the set of definition instructions* ("XML document") *and the set of implementation instructions* ("constraint model") (see at least section 3.1.1, particularly "ensure that your constraint model syntax is supported by the binding framework you want to use" and "Write several XML documents ... and validate them against your new constraints.") McLaughlin further teaches validating <u>*by*</u> *determining whether the class is in* *compliance with the interface* (see at least section 3.4.3 "Verifying Output", particularly: "To ensure that the generated classes work, all you need to do is ensure that they compile"). Applicant teaches that the final validation is done through a compiler in at least page 4 lines 4-34, particularly: "The validity of these interfaces and classes is then proved using methods as described above, i.e. using a compiler that performs the usage and implementation checks at the semantics level." McLaughlin also teaches validating before runtime and during compilation, in at least figure 3-1, pages 13 and 14 "Class generation process flow", particularly "Java compiler."

McLaughlin further teaches *determining whether the method of the interface can be used to execute the runtime function when the object is called during runtime execution of the computer program* (see at least section 3.4.3 "Verifying Output", particularly: "To ensure that the generated classes work, all you need to do is ensure that they compile"). Applicant teaches that the final validation is done through a compiler in at least page 4 lines 4-34, particularly: "The validity of these interfaces and classes is then proved using methods as described above, i.e. using a compiler that performs the usage and implementation checks at the semantics level." McLaughlin also teaches validating *before runtime and during compilation*, in at least figure 3-1, pages 13 and 14 "Class generation process flow", particularly "Java compiler."

McLaughlin teaches a standard Java compiler to validate implementation and definition instructions. McLaughlin does not explicitly teach that the compiler checks the class for features or the interface for methods. However, Gosling teaches a standard Java compiler *determining whether* the object in a first class *is relying on a feature of* a provider class that is not promised by the class (see at least page 347, particularly: "If the class or interface has no method declaration that is both applicable and accessible, then a compile-time error occurs.").

Gosling further teaches *determining whether the interface includes an additional method that is called during runtime execution of the computer program* (see at least page 199: An interface declaration introduces a new reference type whose members are classes, interfaces, constants and abstract methods," emphasis added). As noted in the above rejection under 35 USC 112, first paragraph, under the principles of compact prosecution, examiner treats this limitation as pertaining to the interface promising multiple methods that may be called by a computer program during run time, as is well known in the art of computer programming and partially supported by the cited portion of the specification (page 2 lines 28-31).

Gosling further teaches *verifying whether implementation of the provider class is in compliance with a promise of the provider class, the verification being performed by using the interface* (see at least pages 230, section 12.1.2= and 233-236, section 12.3, "Linking of Classes and Interfaces," particularly: "Verification of the binary representation" and "Verification ensures that the binary representation of a class or interface is structurally correct." and "that every method is provided with a structurally correct signature"). The signature of a method in an interface or class is a promise of that interface or class.

It would have been obvious to one of ordinary skill in the art at the time the invention was made that the standard Java compiler verifications as suggested by McLaughlin include the verification features taught by Gosling because the "Java Language Specification" describes the Java language (Gosling page xix, "This book attempts a complete specification of the syntax and semantics of the language.").

McLaughlin further teaches *wherein the determination is made before runtime execution of the computer program and during compilation of the computer program* (see at least figure 3-1, pages 13 and 14 "Class generation process flow", particularly "Java compiler").

McLaughlin further teaches a computer program comprises other web services (see at least section 1.2.2, particularly "web services"). However, the combination of McLaughlin and Gosling does not explicitly teach a script code section. Lucas teaches *a script code section* (see at least column 3, lines 45-55, particularly "XML-oriented language extensions for use in association with a scripting language").

Lucas further teaches *generating a symbol table based on the script code section, the symbol table including a variable used by the script code section* (see at least figures 3A to 3C and associated text such as 3:58 through 4:7, particularly: "a Javascript-aware parser (e.g. parser 105) is equipped to recognize XML data type declarations and associate them with the appropriate items in the corresponding symbol table"). The association of a variable with an item in the respective symbol table indicates that the symbol table was generated.

Lucas further teaches *validating the script code* ("JavaScript") *section by comparing the symbol table with the set of implementation instructions* ("XML data type declarations") (see at least figures 3A to 3C and associated text such as column 3, line 56 through column 4 line 7, particularly: "a JavaScript-aware parser (e.g. parser 105) is equipped to recognize XML data type declarations and associate them with the appropriate items in the corresponding symbol table").

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the constraint model of McLaughlin with the script code validation of Lucas because it would allow XML constraints to be used and tested with a script-based web service (see at least McLaughlin section 1.2.2 and Lucas column 3 lines 45-55).

Claim 2

Claim 2 includes all of the limitations of claim 1. Applicant states in page 3 lines 6-26 that their program is programmed using two separate tree structures, written in XML, wherein the first tree structure represents the classes to be implemented (definition module) and the second tree structure represents the associated interfaces (implementation module). McLaughlin further teaches *the set of definition instructions are definition modules* (see at least section 2.3.1, particularly "XML file" and figure 2-2 "XML documents"). McLaughlin further teaches *definition modules having a plurality of classes* (see at least section 3.3, particularly: "Once you've got your constraints ... you're ready to create a binding schema for your classes. This will instruct the class generation tool to generate classes, to use a specific Java package, to use collections, and a variety of other options." emphasis added).

McLaughlin further teaches *the set of implementation instructions are implementation modules* (see at least section 2.3, particularly "XML constraints" and section 6.4.4 "Interfaces, particularly: "To generate an interface, add this statement to your binding schema"). McLaughlin further teaches *implementation modules having a plurality of interfaces* (see at least section 3.2, "Creating the constraints", particularly "a set of constraints ready to generate classes from" and example 3-1 "ELEMENT movies" and "ATTLIST movies version"). Because XML constraints are a model of the behavior of classes ("XML document"), they are an interface, as supported by specification page 3 lines 6-26.

Claim 3

Claim 3 includes all of the limitations of claim 1. Claim 3 has been amended to recite "the class having the object is converted into a new class." As noted in the rejection of claim 3 under 35 USC 112, first paragraph for lack of written description, under the principles of compact prosecution, examiner treats this limitation as identical to claim 3 as originally filed. The Examiner understands the original claim 3 to be described in the specification at page 4 lines 4-10, particularly: "the generation of executable programs in script languages, such as for example JavaScript is done by first generating an intermediate code, using a language that supports classes and interfaces."

McLaughlin further teaches *the set of definition instructions are converted into classes* (see at least section 3.1.4, figure 3-1 "Class generation process flow" and "The result of the generation step is one or more Java source files").

McLaughlin further teaches *the set of implementation instructions are converted into interfaces* (see at least section 6.4.4 "Interfaces", particularly "The result of this statement is a new generated class, the Person interface").

Claims 4, 6, 10, and 11

Claim 4 includes all of the limitations of claim 3. Claims 6, 10 and 11 include the limitations of claim 1. McLaughlin further teaches *the set of definition instructions* ("XML documents") *and the set of implementation instructions* ("XML constraints") *are described in XML* (see at least section 2.3.1 , particularly "XML file" and figure 2-2 "XML documents" and section 2.3, particularly "XML constraints"). It is readily apparent that McLaughlin teaches the files are in a tree structure because the files are XML files.

Claim 5

Claim 5 includes all of the limitations of claim 4. McLaughlin further teaches *the class having the object and the interface having the method are defined in Java language* (see at least section 3.1.4, figure 3-1, particularly: "The result of the generation step is one or more Java source files").

Claim 7

Claim 7 includes all of the limitations of claim 1. McLaughlin further teaches a computer program comprises other web services (see at least section 1.2.2, particularly "web services"). However, McLaughlin does not explicitly teach a script code section. Lucas teaches *the script code section is JavaScript* (see at least column 3, lines 45-55, particularly "a scripting language, such as JavaScript"). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the constraint model of McLaughlin with the JavaScript script code validation of Lucas because it would allow XML constraints to be used and tested with a script-based web service (see at least McLaughlin section 1.2.2 and Lucas column 3 lines 45-55).

Claim 8

Claim 8 includes all of the limitations of claim 1. McLaughlin further teaches *validating the script code section comprises generating a symbol table by executing the code section in an* *interpreter* ("parser"), *and comparing the symbol table with the implementation instruction* ("XML data type declarations") (see at least column 3, line 56 through column 4 line 7, particularly: "a JavaScript-aware parser (e.g. parser 105) is equipped to recognize XML data type declarations and associate them with the appropriate items in the corresponding symbol table"). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the constraint model of McLaughlin with the script code validation of Lucas because it would allow validating a script-based web service with XML constraints (see at least McLaughlin section 1.2.2 and Lucas column 3 lines 45-55).

Claims 12-21

Claims 12-21 are a computer readable medium version, which otherwise recites the same limitations of the claims 1-8 and 10-11. The combination of McLaughlin and Lucas teaches all of the limitations of claims 1-8 and 10-11. It is readily apparent that the method taught by McLaughlin includes instructions to implement the method on a computer readable medium (see, for example, section 3.1 step 4, "compile the classes").

Cited Prior Art

28. Hammerich et al. (US PG-PUB 2004/0123273 A1, hereafter '273) was cited in the Office Action mailed 8/28/2009 as claiming priority to the same European application (EP 02022042.2). Hammerich et al. (US 7,584,457) was cited in the Office Action mailed 8/28/2009 as being a granted patent from application '273.

29. **Examiner's Note**: The Examiner has pointed out particular references contained in the prior art of record within the body of this action for the convenience of the Applicant. Although the specified citations are representative of the teachings in the art and are applied to the specific limitations within the individual claim, other passages and figures may apply. Applicant, in preparing the response, should consider fully the entire reference as potentially teaching all or part of the claimed invention, as well as the context of the passage as taught by the prior art or disclosed by the Examiner.

Conclusion

20. Any inquiry of a general nature or relating to the status of this application or concerning this communication or earlier communications from the Examiner should be directed to **Erika Kretzmer** whose telephone number is **(571) 270-5554.** The Examiner can normally be reached Monday through Thursday, 9:30am-6:00pm Eastern Time. If attempts to reach the examiner are unsuccessful, the Examiner's supervisor, **Tuan Dam** can be reached at **(571) 272-3695**.

21. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://portal.uspto.gov/external/portal/pair . Please direct questions on access to the Private PAIR system to the Electronic Business Center (EBC) at **866.217.9197** (toll-free).

22. Any response to this action should be mailed to:

Commissioner of Patents and Trademarks
Washington, D.C. 20231

or faxed to 571-273-8300. Hand delivered responses should be brought to the United States Patent and Trademark Office Customer Service Window:

Randolph Building
401 Dulany Street
Alexandria, VA 22314.

/Erika Kretzmer/
Examiner, Art Unit 2192
2/8/2011

/Michael J. Yigdall/
Primary Examiner, Art Unit 2192